Chapter 4 – Web Agents for Supporting Collaborative Learning

This chapter explores the field of web agents and attempts to utilize these agents to support the collaborative learning activities. The first part of this chapter discusses the roots of web agents through theoretical aspects of software agents such as agent definitions and classifications. This is followed by the discussion of web agents that includes its definition, deployment potentials and types.

Furthermore, this chapter discusses a review of three existing educational applications on their agent architecture that implement web agents with different roles to assist student's learning process and investigate an architectural scheme of a Multi-Agent System for distributed collaborative learning environment. Next, this chapter discusses web agents' potentials in supporting collaborative learning activities. Lastly, it focuses on web agents to support G-Jigsaw collaborative learning.

4.1 Agent Definitions

According to Alan Kay, the idea of agent was originated from John McCarthy in the mid-1950s and Oliver G. Selfridge coined the term "agent" a few years later, when they were both at the Massachusetts Institute of Technology (Kay, 1984). Although the term has been widely used for almost 20 years, yet there is not a single universal accepted definition to date (Wooldridge and Jennings, 1995; Nwana, 1996; Bradshaw, 1997).

In explaining why it is so difficult to define precisely what agents are, Nwana (1996) has pointed out two major reasons that defy attempts to produce a universal accepted definition for agent. Firstly, the term "agent" is not only being used in Artificial

Intelligence field, but it is also used widely in everyday parlance as in travel agents and estate agents. Secondly, even within the software fraternity, the word agent is really an umbrella term for a heterogeneous body of research and development.

Due to the lack of a universal accepted definition for the term agent, researchers in agent arena invented yet more synonyms including knowbots (i.e. knowledge-based robots), softbot (software robot), taskbots (tasks-based robot), userbots, robots, personal agents, autonomous agent and personal assistants (Nwana, 1996). Bradshaw (1997) also claimed that varieties of 'Agents' have proliferated. There has been an explosion in the use of the term without a corresponding consensus on what it means. In his book entitled "Software Agents", he provides many examples of claimed 'agents' but with different meanings.

Even though there is yet to have a consensus definition for 'agent' at present, a general agent definition can be obtained by summarizing several definitions provided by the researchers working in the field of agent. Based on this general definition, an agent is defined as an entity (software/program and/or hardware) that performs some tasks or set of tasks on behalf of its user within a computer environment such as operating systems, databases, or computer networks (Andoh et. al., 2001; Yahya, 2001; Baylor, 1999; Nwana, 1996).

Along with this line, Olguin and colleagues (2000) regard an accepted definition of an agent as "An agent is a computational entity that (i) executes in behalf of other entities (users, programs, etc) in an autonomous way; (ii) makes actions in a pro-active and/or a reactive way; and (iii) presents some capabilities to learn, cooperate and move" (Olguin et. al., 2000). For more agent definitions, refer Franklin and Graesser's (1996) paper entitle “Is it Agent, or just a Program?: A Taxonomy for Autonomous Agents”.

4.2 Agent Classifications

The classification of agents provides a simpler way of characterizing the various types of agents. There are many classifications of agents such as Wooldredge and Jenning’s agent notation (Wooldredge and Jenning, 1995), Nwana’s agent typology (Nwana, 1996), Franklin and Graesser’s taxonomy of agent (Franklin and Graesser, 1996) and Bradshaw’s classification schemes and taxonomies of agent (Bradshaw, 1997). This chapter discusses Nwana’s agent typology with an attempt to determine which category should the web agents belong to.

4.2.1 Nwana’s Agent typology

A typology refers to the study on types of entities (Nwana, 1996). Nwana proposes a typology of agents that attempts to classify most of existing software agents into different agent classes. These agents are categorized according to five major dimensions.

The first dimension categorizes agents by their mobility. Mobility means the agents ability to move and travel around the network. This dimension yields the classes of static or mobile agents. Mobile agents are computational software process capable of wandering wide area network, interacting with foreign hosts, gathering information on behalf of their users and return home upon completing their tasks. In contras, static agents are agents that perform their tasks in a local environment without the capability of moving.

The second dimension classifies agents as either deliberative or reactive agents. Deliberative agents possess an internal symbolic, reasoning model and they engage in planning and negotiation in order to achieve coordination with other agents (Nwana 1996). On the other hand, reactive agents do not have any internal, symbolic models of their environment. They act using a stimulus/response type of behavior by responding to the present state of environment in which they are embedded (Ferber, 1994).

The third dimension compartmentalizes agents along several ideal and primary attributes that agents should reveal. Nwana and his colleagues have identified three attributes, namely autonomy, learning and cooperation. Autonomy refers to the principle that agents can operate by themselves without the need of human guidance. Therefore, agents with this attribute contain individual internal states and goals and they perform on behalf of its user to meet their goals.

Cooperation with other agents is paramount. In order to cooperate, agents need to possess a social ability, which means the ability to interact with other agents and possibly humans via some communication language (Wooldridge and Jennings, 1995). Agents also need to disembody bits of intelligence via learning attributes. Agents would have to learn as they react and/or interact with their external environment to enhance their performance over time.

Nwana and his colleagues use these three minimal attributes to derive four more types of agents to include in their typology, which are collaborative agents, collaborative learning agents, interface agents and truly smart agents. Collaborative agents emphasize more on cooperation and autonomy, collaborative learning agents stress more on cooperation and learning, whereas interface agents focus more on autonomy and learning attributes. Truly smart agents have all the three attributes, which are autonomy, cooperative and learning.

However, these distinctions are not definitive. If collaborative agents inherit a combination of cooperation and autonomy attributes that does not imply that this agent will never learn. It only shows that collaborative agents emphasize more on cooperation and autonomy rather than learning. Figure 4-1 below shows Nwana's Agent Typology.

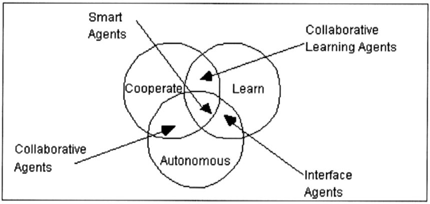

Figure 4-1 Nwana's Agent Typology (from Nwana, 1996)

The fourth dimension groups the agents according to their major roles. A typical example of these types of agents is WWW information agents. This category of agents usually exploits Internet search engines (e.g. WebCrawlers, Lycos and Spiders). These types of agents sometimes are referred as Internet/Information agents. Essentially, they assist in managing the vast amount of information in Wide Area Network, Local Area Network and Internet.

The last important dimension is a special form of categorization of agents. This dimension classifies agents that combine two or more agent philosophies in a single agent, namely the hybrid agents.

Other attributes such as versatility, benevolence, veracity, trustworthiness, temporal continuity, ability to fail gracefully; and mentalistic and emotional qualities are considered as secondary attributes in this typology (Bradshaw, 1997).

From the typology, Nwana and his colleagues pointed out that they did not find any agents that could fit the collaborative learning agents and truly smart agents' descriptions. Hence, only 7 types of agents are identified in this typology: Collaborative agents, Interface agents, Mobile agents, Information/Internet agents, Reactive agents, Hybrid agents and Smart Agents. These categories of agents are discussed in the subsequent section.

4.2.2 Different Agent Types Identified by Nwana

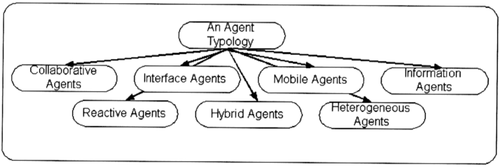

Figure 4-2 A Summary of Nwana's Different Agent Types (from Nwana, 1996)

As mentioned in section 4.2.1, there are all-together 7 different agent classes identified in Nwana's typology. Figure 4-2 above summarizes all the agent types in Nwana's typology. This section briefly overview each type of these agents. For more detailed information on these agents such as their metaphors, hypothesis/goals, motivations, roles, prototypical examples, potential benefits, key challenges and some other general issues, refer to Nwana's (1996) paper "Software Agent: An Overview".

4.2.2.1 Collaborative Agents

Collaborative agents emphasize on autonomy and cooperation while carrying out operations for their owners. Even though these agents have the ability to learn, however most of them do not perform any complex learning. Some researchers provide stronger definitions such as beliefs, desires and intentions or emotional attributes to these agents. Collaborative agents must have the ability to negotiate in order to reach an equally acceptable consensus when carrying out their operations. Autonomy, social ability, responsiveness and pro-activeness are the key characteristics of these agents. As a result, they are able to act rationally and autonomously in open and time-constrained multi-agent environments.

4.2.2.2 Interface Agents

Interface agents emphasize on autonomy and learning in order to assist their owners to perform their tasks. As noted by Pattie Maes, the underlying key metaphor of interface agents is that it is a personal assistant who collaborates with users in the same work environment. Interface agents usually support and provide users with assistance in using a particular application. Essentially, operations performed by these agents include observing and monitoring user's actions, learning new shortcuts and proposing better ways of doing a task. (Maes, 1994) has reported that interface agents learned by observing and imitating the user, receiving user's positive and negative feedback, receiving user's explicit instructions and inquire advice from other agents.

4.2.3.3 Mobile Agents

Mobile agents are computational software process that is capable of traveling around wide area network, interacting with foreign hosts, collecting information on behalf of its owner and return home upon completing its tasks. Although mobile agents inherit the autonomy and cooperation attributes but they are distinguish from collaborative agents. This is because they are able to exchange only requested information with other agents without exposing other information. In other words, mobile agents may communicate or cooperate with one agent to make some of its internal objects and methods available for other agents.

4.2.2.4 Information Agents

Unlike other types of agents described previously, information or Internet agents are defined by their role, not by their attributes. In other words, they are defined by what they do, in contras with other agents such as collaborative and interface agents, which are defined by what they are. Information agents are created due to the great demands for a tool to assist users in managing the explosive growth of information. Generally, these agents perform the role of managing, manipulating and gathering information from many distributed sources. Interface agents that are employed with WWW-based roles are also called information agents.

4.2.2.5 Reactive Agents

Reactive agents, in contras with deliberative agents, do not possess internal and symbolic models of their environments. They behave in a stimulus-response manner towards the current state of the environment where they are embedded. Although this type of agents is relatively simple and only interact with other agents in basic ways, however when they are viewed globally, these interactions appeared to be a complex patterns of behavior. (Maes, 1991) emphasizes three key ideas that underpin reactive agents which include the emergent functionality (i.e. the dynamic of interaction leads to the emergent complexity), tasks decomposition (i.e. a reactive agent is viewed as a collection of modules which operates autonomously and is responsible for specific tasks) and the tendency of reactive agents that operate on representation that are close to raw sensor data.

4.2.2.6 Hybrid Agents

Hybrid agents refer to those agents that are formed using the combination of two or more agent philosophies (such as mobile philosophy, interface agent philosophy and/or collaborative philosophy) within a singular agent. Since each type of agents has their own strengths and deficiencies, therefore, the hybrid approach is adopted with the goal to maximize the strengths and at the same time trying to minimize the deficiencies for a particular purpose. As a result, the key concept of implementing hybrid agents is based on the belief that for some application, benefits gained from using a singular agent with combination of philosophies are greater that using an agent with a singular philosophy.

4.2.2.7 Heterogeneous Agents

Heterogeneous agents system refers to an integrated setup of at least two or more agents, which belong to two or more different agent types. This type of agents differ from hybrid agents because it consists of two or more agents from different types that work independently yet collaboratively in an integrated environment, whereas hybrid agents try to combine two or more different agent philosophies within a singular agent. The motivation of this type of agents was articulated by Genesereth & Ketchpel (1994), that the world abounds with a rich diversity of software products providing a wide range of services for a similarly wide range of domains. Though these programs work in isolation, there is an increasing demand to have them interoperate - hopefully, in such a manner such that they provide added value as an ensemble than they do individually.

4.3 Web Agent

In conjunction with the vast growth of the Internet and WWW, web agents are getting more and more important roles in software agent research and development that resided on various domains. In Sean Luke and James Hendler's (1997) introductory paragraph, they gave a good illustration on the emergence of web agents in the near future.

> "*There are two kind of information-seeker currently wandering the World Wide Web. First there are us humans, the web surfers for whom the web was designed. Second, there are increasing numbers of automated systems, Web Agents, which gather information from the web on our behalf. At the present time, humans far outnumber web agents, but this could soon change: as the sheer volume of information on the Web increases, and the ratio of junk and useful information continues to grow, we will increasingly rely on web agents to dig through all that muck to find our germs for us" (Luke and Hendler, 1997).*

Essentially, web agents are deployed especially in collecting, filtering and manipulating the vast information via the Internet. Indeed, some of these agents even exploit artificial intelligence and advance information technology on the Internet and WWW. Intelligent web agents offer a profusion of useful features such as provide a convenient graphical user interface to present information in a Web browser, securely handle users' Web site and autonomously perform various repeating tasks on behalf of the users.

Fundamentally, web agents are some automated programs, which perform tasks of gathering, clustering and filtering information from the web on behalf of their users. Besides, there are also some intelligent web agents that assist the users to browse and search through the web, suggesting useful links to help the users to obtain their information easily and timeliness.

Based on Nwana's typology presented in section 4.2, web agents are best suit under the category of Information/Internet Agent (discussed in section 4.2.2.4). However, as time pass by, more and more web agents of other categories (e.g. Interface Agent, Mobile Agent and Reactive Agent) exist. Due to this, web agents should be defined by their role, not by their attributes. In other words, they should be defined by what they do not by what they are.

4.3.1 The Potentials of Web Agents

A recent review on software agents' research has indicated that web agents usage have great potentials in the future. In 1999, Nwana and Ndumu reported that agents started to become an exhortation around 1994 and the agent field is becoming more matured ever since then.

> *"Though "agents" research had been going on for more than a fifteen years before, agents really become a buzzword in the popular computing press (and also within the artificial intelligence and computing communities) around 1994"(Nwana and Ndumu, 1999).*

In the same paper, they further concluded that this change was due to the growth and explosion of the WWW.

> *"During late 1994 and throughout 1995 and 1996, we saw an explosion of agent-related articles in the popular computing press. It is no coincidence that this explosion coincided with that of the World Wide Web (WWW)."(Nwana and Ndumu, 1999)*

Since information posted in the Internet and WWW are progressively increasing, this opens opportunities to utilize the web agents. Furthermore, an independent telecommunication, new media and information technology analyst group named Ovum Inc. produce a report titled “Intelligent agents: the new revolution in software”. In this report, they wildly speculated the total market sector for software agent and products by the year 2000. It predicts a $4 billion software agent market in the year 2000 with applications of agent technology appearing in the computing, telecommunications, consumer, entertainment, manufacturing and military market segment. Besides, Ovum anticipates that the Internet will support 500 million users worldwide by 2005. The findings are available in a new report from Ovum, Internet Market Forecasts: Global Internet Growth 1998 – 2005. Hence, in the next five years, agent-based technologies and services will become common. The growth in Internet and WWW will encourage towards the use of web-based agents.

4.3.2 Types of Web Agents

As web agents come in various shapes and size, Luke and Hendler have identified three categories of web agents: "Off-line" agents, "On-line" agents and "Guide" agents. "Off-line" agents gather all the information they possible find on Web, then allow users query this information according to their needs later. This type of agents is usually in the form of text indexing search engines like Lycos & AltaVista. Conversely, "On-line" agents search the web with a query in-hand. ShopBot search engine is an example of on-line agents. Lastly, "Guide" agents work alongside the users, helping the users focus their browsing in real time as they search for information on their own. These agents act as tour guide and steer the users along an appropriate path through the collection, based on its knowledge of user's interests, of the location and relevance of various items in the collection, and of the way in which others have interacted with the collection in the past.

In addition, there are many web agents exist in various web applications. Besides the three general categories of web agents (i.e. "Off-line" agents, "On-line" agents and "Guide" agents) that have been identified by Luke and Hendler, these web agents can be further categorized into more specific types of agent according to their roles. These agents include information filtering agents (e.g. NewsWeeder (Lang, 1995) and Webhound (Lashkari, 1995)), information search assistant agents (e.g. WebWatcher (Joachims et. al., 1997), LAW & ELVIS (Edwards et. al., 1997) and Syskill & Webert (Pazzani el. at., 1996)), information clustering agents (e.g. CUSTARD (Edwards et. al., 1997)), information advising agents (e.g. MAVA (Edwards et al., 1997)), collaborative web agents (e.g. Do-I-Care (Starr et. al., 1996)), personal agents (e.g. Remembrance Agent (Rhodes and Starner, 1996)) and more recently, the interactive assistance agents (e.g. Microsoft Agent). Most of these web agents have been embedded into different web applications to perform various tasks on behalf of their users.

Information filtering agents such as NewsWeeder and Webhound, as they are named, are performing the tasks of filtering information from the Web and presenting only the information that the users selected or the information that may of interest to the users. NewsWeeder enables the users to personalized list of article summaries from selected group of articles while Webhound recommends new documents to the users based on observation and employs documents to a user by comparing materials deemed to be of interest to one user with database of other user preferences.

Information search assistant agents such as WebWatcher, LAW & ELVIS and Syskill & Webert normally attempt to recommend links that a user should follow, help a user to find new and interesting information on the Web or deciding what pages might be an interest to the user. WebWatcher recommends links to its user, observes user's reactions towards its advice so that it could eventually predicts a user's action. LAW interactively suggests links to the user as they browse the Web and uses a separate Web robot that attempts to find pages that might be an interest for the users while Syskill & Webert learns a user profile by analyzing information on a page. It uses the user profile to suggest links and construct a query to find pages that the user would be interested to explore.

Other agents carry out their tasks based on their roles. Information clustering agents like CUSTARD groups similar documents together and presents the user with clusters of documents, rather than a flat list. MAVA, an information-advising agent, employs a number of agents to mediate access to a collection of databases. Do-I-Care is an innovative WWW agent that collaborates with both users and their peers to identify potential interesting changes. It works by soliciting user opinions about changes it finds to train a user model. Recently, Microsoft has developed interactive assistant agents named Microsoft agents, which is a set of programmable software services that supports a presentation of interactive animated characters within the Microsoft Windows interface. These agents work as interactive assistants to introduce, guide, and entertain or enhance Web pages or applications. They enable the incorporation of conversational interfaces, which leverages natural aspects of human social communication. In addition, Microsoft agents support speech recognition, hence an application will have the capability to respond to voice commands.

Based on information gathered above, the author formulates a diagram to illustrate various types of web agents with different roles on the Internet as shown in Figure 4-3.

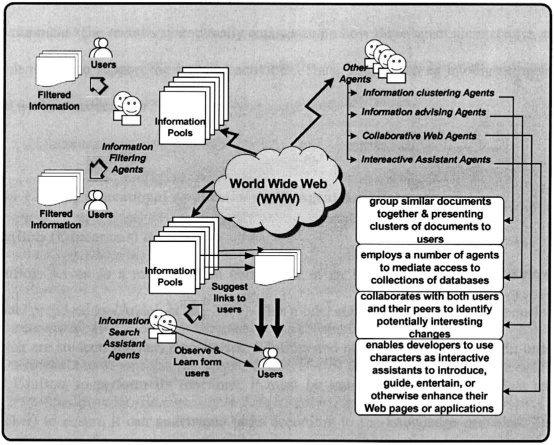
Users
Filtered Information
Information Pools
Information Filtering Agents
Filtered Information
Users
World Wide Web (WWW)
Other Agents
Information clustering Agents
Information advising Agents
Collaborative Web Agents
Intereactive Assistant Agents
group similar documents together & presenting clusters of documents to users
employs a number of agents to mediate access to collections of databases
collaborates with both users and their peers to identify potentially interesting changes
enables developers to use characters as interactive assistants to introduce, guide, entertain, or otherwise enhance their Web pages or applications
Information Pools
Suggest links to users
Information Search Assistant Agents
Observe & Learn form users
Users

Figure 4-3 Various Types of Web Agents with Different Roles on the Internet

4.4 Agent Architectures for Educational Applications

Even though the implementation of web agents in educational applications is currently not a norm yet, however, there are indications that most future educational applications would implement various web agents to accomplish complicated tasks. At the time of writing, the author does not found any collaborative learning applications that implemented web agents in supporting the collaborative learning process. As a result, section 4.4.1 reviews four agent architectures that implemented web agents in their educational applications. Three of these agent architectures have been implemented in the existing applications namely EduBots (Andoh et. al., 2001), CALM (Olguin et. al. 2000)and ANGEL (Jafari, 1999). The remaining agent architecture (Fenton-Kerr et. al., 1998) is a proposal of an architectural scheme for distributed collaborative learning environment. The reviews intentionally emphasize on how these agent architectures are implemented to support the learning activities. Thus, issues such as intelligent agents will not be considered in detail.

4.4.1 Existing Educational Applications with Agent Architectures

EduBots (Educational Robots)

EduBots serves as a major system component in an Intelligent Educational System model proposed by (Andoh et. al., 2001). This model consists of five basic components, which are student, teacher, repositories, EduBots and communication system. In order for EduBots to perform its functions, it must be trained by the users (student and teacher) to ensure it can understand tasks according to the knowledge provided. The EduBots will store and keep provided knowledge into its knowledge base. Once these Edubots are trained, student and teacher can request Edubots to perform tasks such as analysis courseware, library and assessment repositories and providing responses according to its knowledge base. The responses can be sent through communication system or displayed on web browser.

To support the educational system, EduBots applies multi-agents which are interface agent, assessment agent, email agent and sharing agent. Interface agent is a web-based graphical interface where it enables users to interact with Edubots. The primary functions of the interface agent are to execute the assessment and the sharing agent, to display information to the users and to communicate with other agents and with the knowledge base. Email agent is responsible for generating, composing, organizing and sending emails to both students and teachers. The assessment agent analyzes assessment repository and determines which part of learning materials should be re-learned. The sharing agent is responsible for checking library repository for users where it can determine which articles or documents in the library repository are useful for its users based on the pre-defined user's requirements knowledge. The internal architecture of EduBots is depicted in figure 4-4.

Andoh and his colleagues stated several potential advantages of EduBots in educational system. These include sharing knowledge effectively, enabling cooperative task resolution among users, providing dynamic and personalized interface, improving the distribution of modules as well as communication among users.

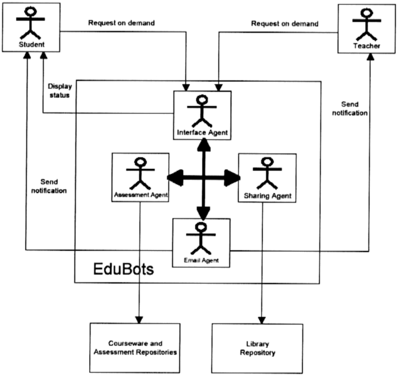

Figure 4-4 EduBots' Internal Architecture (Adapted from Andoh et. al., 2001)

CALM (Computer Aided Learning Material)

CALM is an on-line learning environment which is developed at UNICAMP. It is developed to recommend lessons regarding a student's learning objective and profile and to support interaction among actors and a learning material (Adriano et. al., 1999). CALM can be viewed as a kind of recommender system that work on sets of texts to assist and augment the social process of recommendation. In such process, a set of recommendations on a given subject (e.g., URL's in a newsgroup) is aggregated and delivered to an appropriate destination (a person or a repository). The main characteristic of these systems is the ability to choose and classify recommendations from input, based on weighted voting and content analysis. CALM differs from the traditional recommender systems that it regards to subjects already done by students, learning materials available, and the subjects that aimed to be learned. CALM recommendation is a composition that comprise of the desired subjects and a set of recommended subjects that are supposed to be known.

Olguin and colleagues proposed an Agent-based Architecture for group study in 1999. There are four agency working together in this architecture namely Group Agency, User Agency, Activities Agency and Advisory Agency. Figure 4-5 shows the Agent Architecture proposed by Olguin. In this architecture, the Group Agency performs tasks related to search and invitation processes. The User agency is associated with any user in the learning environment, group owner or not. This agency is responsible to interact with search agents sent by Group agencies.

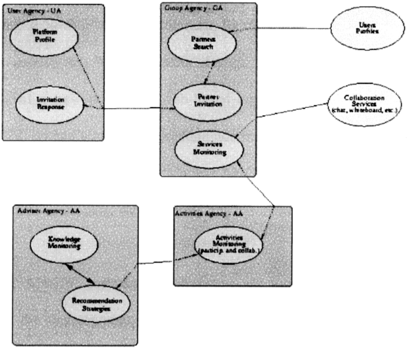

Figure 4-5 Agent Architecture (Adapted from Olguin et. al. 2000)

The Activity Agency uses the information collected by Group Agency to generate more information that is important for reasoning group members' activity and participation levels, and evaluating the effectiveness of a group. The Advisor agency performs most of the tasks related to group and members knowledge monitoring. It enables the customization of generated recommendations send by the group advisor based on the members profile and information related to the group activities. This agent-based architecture is implemented as a service of CALM system. It only implements the Group and User Agencies, which using CALM as a test-bed.

ANGEL (A New Global Environment for Learning)

ANGEL, A New Global Environment for Learning, is a web-based enterprise course management system, which is a further enhancement of the Oncourse project (Jafari, 1999). With the successful achievement of Oncourse, CyberLearningLabs of Indiana

University continued their Oncourse research and development project in the direction of developing a new global environment for learning. The objectives of ANGLE include the conceptual design, system design, prototype development and system development of a new intelligent web-based environment for teaching and learning needs. ANGEL project's major features include intelligent agents, enterprise framework and distributed authentication. Additionally, ANGEL focuses on theoretical research in the area of cognitive science, web usability and assessment.

The goals of ANGEL software development are focused on applying the research findings into the development of a complete software enterprise system. ANGEL's goals are broader than the developed Web-based "courseware" solutions. ANGEL supports standards for interoperability which is currently in development by the Internet Engineering Task Force and IMS EDUCAUSE and creating global solutions for teaching and learning. These software solutions are developed as modules, making the environment adaptable to the needs of educators of various resources. Utilizing IMS Metadata protocols, the objects in ANGLE environments (which may consist of software resources, course content, video or audio resources, etc.) are readily searchable and accessible through traditional search engines, but with more concise query methods. This global access is managed by the authentication system, allowing resources to be made private, local or global depend on the author's preferences.

Moreover, the ANGEL environment includes a series of intelligent agents to provide a more effective, interactive and dynamic environment for teaching and learning. ANGLE's architecture is designed in a way that it can be modified and extended since e-learning strategies change and grow. Intelligent agents are one type of the extension.

Generally, there are five intelligent agents developed in ANGLE, namely Who Dun It Agent, What's New Agent, Learner Profile Agent, PDA Agent and Ungraded Items Agent. The Who Dun It agent is responsible to generate course mail to specific users based on their activity. For instance, it can be used to find students who scored above 75% on a quiz and send an email to each of them. The What's New agent searches items that have not been visited or completed by the user and brings them to the user's attention. This may helps the user to catch up quickly. The Learner Profile Agent enables the assessment of every student's total performance. It reduces the time spent in assessing student performance and participation, provides information on every aspect of each student's performance and relates each student's performance to his/her class. The PDA Agent provides access to ANGLE at any time and place. It enables user to download news, calendar and unread course item into the PDA (Personal Digital Assistant). The Ungraded Items Agent plays the role of finding and listing all items with one or more ungraded submission to ensure action can be carried out accordingly.

Architectural Scheme for Distributed Collaborative Learning Environment

Fenton-Kerr's architectural scheme is proposed for supporting the distributed collaborative learning environment (Fenton-Kerr et. al., 1998). Past multi-agent schemes are usually implemented in a situation where net-based transactions will occur. The typically agents in such multi-agent schemes include interface agents, informational agents and coordinating or planning agents (Fenton-Kerr et. al., 1998). To extend this concept, Fenton-Kerr proposed 3 additional entities specifically designed for a distributed collaborative learning environment. The 3 extended entities are LOCAL agent, TRACKER agent and DICTIONARY agent.

The LOCAL agent has local knowledge, but not specialized knowledge. It acts as a guide to modules, explaining the actions of various screen objects such as input and result fields and buttons. The TRACKER agent tracks a user's passage through the program, makes inferences on the user's current level of understanding (i.e. basic user modeling) and keeps a record of the user's subject-specific input. It may also maintain a log of user's previous sessions within-course assessment. Finally, the DICTIONARY agent offers contextual explanations or definitions. It may make use of alternative input modes such as audio-based requests, or gestural (deictic) responses (such as circling a word). In the proposed approach, a variation on the federated system (Genesereth and Ketchpel, 1994), detailed in figure 4-6, the coordinating agent plays the role of a facilitator.

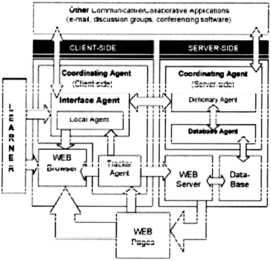

Figure 4-6 A Proposed Architectural Scheme of a Multi-Agent System (Adapted from Fenton-Kerr et. al., 1998)

As shown in figure 4-6, the Database Agent retrieves database information by developing appropriate structured query statements. As the only agent with knowledge of database and its contents, it mediates requests from other agents such as Dictionary

Agent and system-level requests from the web server. The Tracker Agent has two tasks. It keeps track of the current page contents and logs a user's entire session, including visited pages, test results or other interactions.

4.4.2 Summary on reviewed Agent Architectures

Table 4-1 provides a summary of web agents' implementation in education applications based on the web agents involved, their roles and characteristics, as well as its applied architecture in each application.

Table 4-1 Summary of Web Agents' Implementation in Educational Applications

Application	Agents Involved	Roles & Characteristics	Architecture
Edubots (web-based), 2001	Interface Agent	- To executive the assessment agent & sharing agent - To display information to users - To communicate with other agents and with knowledge based	Collaborative Agents Architecture (Multi-Agent System)
	Sharing Agent	- To determine user's interested articles or documents based on pre-defined user's knowledge requirements	
	Assessment Agent	- To analyze assessment repository - To determine which part of material should be re-learned	

	Email Agent	- To be responsible in generating, composing, organizing and sending emails to both the students and teachers	
CALM (web-based) 2000	Group Agency	- To perform the tasks related to search and invitation processes	Agent-based Architecture (Multi-Agent System)
	User Agency	- To interact with search agents sent by Group agencies	
	Activity Agency	- To generate more information for reasoning about activity and participation levels of group members	
	Advisory Agency	- To perform most of the tasks related to group and members knowledge monitoring	
ANGLE (web-based) 2000	Who Dun It Agent	- To generate course mail to specific users based on their activity	ANGLE System Architecture
	What's New Agent	- To search items that have not been visited or completed by the user - To bring these items to the user's attention	

	Learner Profile Agent	- To monitor every student's performance and participation - To maintain every student's activity log - To reduce the assessment time of students' performance	
	PDA Agent	- To provide access to ANGLE anytime, anyplace	
	Ungraded Items Agent	- To find and list all items with one or more ungraded submission	
Fenton-Kerr's architectural scheme	Local Agent	- To explain the actions of various screen objects to the modules	Proposed Architectural Scheme (Multi-agent System)
	Tracker Agent	- To track a user's passage through the program - To make inferences about a user's current level of understanding - To keep a record of a user's subject-specific input - To maintain a log of user's previous sessions	
	Dictionary Agent	- To offer contextual explanations/definitions	
	Database Agent	- To retrieve database information by developing appropriately structured query statements	

	Coordinating Agent	- To transform the application-level messages and route them to the appropriate places	

4.5 Web Agents in Supporting Collaborative Learning

The advancement of Internet and WWW has significant impact on education. The Internet provides an environment where everyone can teach and learn collaboratively. With the WWW, worldwide information and global dimensions of education interaction become feasible. These changes have direct impact on collaborative learning activities. They provide network infrastructures and knowledge sharing spaces for collaborative learning. Under such circumstances, the web agents have great potentials in supporting the collaborative learning as well.

According to Kristensen (2001), one of the major challenges in the field of CSCL is to provide the learners with an environment that helps them relate new knowledge to knowledge they already have internalized, and to integrate this knowledge with knowledge from different information technology tools. Web agents, particularly information agents can be used to perform these tasks by providing the learner access to larger knowledge resources through the WWW, searching and filtering related information on behalf of the learners.

Besides, web agents also can be used to support the students in learning process. Normally, in a collaborative learning environment, the students have access to other students' and teachers' ideas and concepts. Web agents can be used to enable the students to exchange and synthesize viewpoints with many students and teachers. To promote more effective collaboration, web agents such as personal assistant agents can be used to help a student to obtain assistance from other students while he/she performs a task. This surely simplifies the student's learning process.

In addition, collaborative learning requires organization and coordination of people and resources within the distributed environments. Web agents can play a crucial role in the coordination processes. Moreover, some of the collaborative process involved several stages with different activities at each stage. Under such circumstances, web agents can be used to automate the complicated collaboration processes in simplifying the collaborative learning. This can be achieved by controlling the flows of the activities and guiding the students' navigation.

Besides, knowledge sharing is a major requirement in a collaborative learning environment. Creating a shared repository which can be publicly accessed and easily reused normally requires great efforts and time. In this context, web agents are best to be used to administer and maintain the shared repository. These tasks include searching, categorizing and storing the information, and enabling the retrieval of these information in an easy manner.

From the discussion in section 4.5, many potential roles of web agents in supporting collaborative learning are identified. These roles are summarized in table 4-2 below:

Table 4-2 Web Agents Roles in Supporting Collaborative Learning Activities

Types of possible web agents	**Major roles**	**Examples**
Information Agents	Provide access to worldwide knowledge resources, search, filter and integrate related information for its users.	Search Agent, Filtering Agent, Integration Agent, etc.
Personal Assistant Agents	Assist users to get help while they are performing a learning task by enabling the communication among the students	Mentor Agent, Interactive Assistant Agent, Personal Agent,

	and teachers.	etc.
Coordinating and Autonomous Agents	Coordinating the flows of collaborative process and automating some complicated processes.	Navigation Agent, Tracker Agent, Information Advising Agent etc.
Repository or Database Agents	Administer and maintain the large volume of information in the shared repository or database.	Sharing Agent, Filtering Agent, Clustering Agent, Profile Agent, etc.

4.6 Web Agents in supporting G-Jigsaw Collaborative Learning

Form the discussion in this chapter, there are many potentials to deploy web agents to support collaborative learning activities. The author attempts to utilize web agents to automate and simplify the G-Jigsaw activities. Based on the web agent roles identified in section 4.5 and the drawbacks of G-Jigsaw prototype discussed in section 3.8, the following are the potential roles of web agents in supporting G-Jigsaw:

- Sharing Agent – to enable teachers to retrieve and use existing questions for creating a jigsaw task, to enable students to retrieve the responses and summaries for the jigsaw task, as well as to filter, categorize and save new questions in a shared repository automatically.
- Integration Agent – to enable the students to perform the report integration automatically.
- Navigation Agent – to control and automate the flows of G-Jigsaw activities by keeping track of every student's jigsaw task profiles.

The next two chapters will discuss the design and implementation of G-Jigsaw which utilizes the web agents to support jigsaw collaborative learning activity. Chapter 5 will discuss the analysis and design of G-Jigsaw while chapter 6 will describe its web agents' implementation.

4.7 Review Summary

This chapter has presented a thorough literature review on web agents. The fundamental theoretical aspects of software agent such as agent definitions and classifications are discussed. The web agent definition, the potentials of its deployment and the various types of web agents are investigated. Three existing educational applications that implement web agents and an architectural scheme for distributed collaborative learning environment are reviewed. As a result, web agents' potentials in supporting collaborative learning activities are identified. Finally, web agents to support G-Jigsaw collaborative learning are discussed.